

---

## PENERAPAN ALGORITMA *BREADTH FIRST SEARCH* UNTUK RANCANG BANGUN *EDUGAME ICON* PERANGKAT TEKNOLOGI INFORMASI BERBASIS *ANDROID*

Yoel Prasetyo<sup>1</sup>, Satria Wisnu Dewa<sup>2</sup>, Daniel Udjulawa<sup>3</sup>

<sup>1</sup>Program Studi Teknik Informatika, STMIK GI MDP, Palembang

e-mail : <sup>1</sup>[joelprasetyo@gmail.com](mailto:joelprasetyo@gmail.com), <sup>2</sup>[satriabaja@mhs.mdp.ac.id](mailto:satriabaja@mhs.mdp.ac.id), <sup>3</sup>[danieludjulawa@gmail.com](mailto:danieludjulawa@gmail.com)

### Abstrak

Dalam perkembangannya, teknologi informasi sudah menjadi elemen penting dalam kehidupan masyarakat sehari-hari. Namun kebanyakan, masyarakat hanya tinggal memanfaatkan teknologi informasi ini tanpa mengetahui nama ataupun fungsinya. Padahal, dasar-dasar teknologi informasi sangat penting agar masyarakat tidak tertinggal dari kemajuan teknologi informasi yang sudah semakin pesat. Salah satu cara mengatasi permasalahan tersebut dengan dibuatnya suatu *game* edukasi berbasis *android*. Dalam penerapannya, *edugame* ini membutuhkan suatu algoritma untuk pencarian solusi. Algoritma yang digunakan adalah *Breadth First Search*. *Breadth First Search* merupakan pencarian yang dilakukan secara melebar dari kiri ke kanan, dan akan selalumenemukan lintasan terpendek yang pertama. Metodologi yang digunakan dalam penelitian ini adalah metodologi *Microsoft Solution Framework*. Dengan algoritma *Breadth First Search* tersebut, dapat memberikan solusi bagi pengguna dalam proses pencarian *icon*.

**Kata kunci :** *Breadth First Search, edugame, icon, software, hardware, android, Microsoft Solution Framework*

### Abstract

*In its development, information technology has become an important in people's everyday lives. But mostly, people just stay utilize this technology without knowing the name or function. Whereas, the basics of information technology is very important that the community is not left behind on the progress of information technology has been growing rapidly. One way to overcome these problems with the establishment of an educational game based on android. In practice, tihs edugame requires an algorithm to search for solutions. Algorithm used is Breadth First Search. Breadth First Search is a search carried out wide from left to right, and will always find the shortest path first. The methodology used in this study is the methodology Microsoft Solution Framework. With the algorithm Breadth First Search, can provide a solution for users in the process of search icon.*

**Keywords :** *Breadth First Search, edugame, icon, software, hardware, android, Microsoft Solution Framework*

---

## 1. PENDAHULUAN

Teknologi informasi merupakan bidang ilmu yang mempelajari tentang perangkat-perangkat informasi baik itu perangkat lunak maupun perangkat keras yang berfungsi untuk mengolah dan menghasilkan informasi. Memang tak dapat dipungkiri lagi, kehidupan masyarakat sekarang tak lepas dari teknologi informasi. Kebutuhan akan teknologi informasi ini tidak memandang status maupun usia. Namun kebanyakan, masyarakat hanya tinggal memanfaatkan teknologi informasi ini tanpa mengetahui nama ataupun fungsinya. Misal, ketika ditanya apa fungsi *Disk Operating System* (DOS), kebanyakan masyarakat tidak mengetahuinya. Ini dikarenakan kurangnya pembelajaran dasar-dasar teknologi informasi bagi masyarakat. Padahal dasar-dasar teknologi informasi ini juga penting agar masyarakat tidak tertinggal dari kemajuan teknologi informasi yang sudah semakin pesat.

Kemajuan teknologi informasi juga diikuti dengan kemajuan industri *game*. Salah satu yang sedang melambung tinggi saat ini adalah *game* dari *smartphone* yang berbasis *android*. Aplikasi *game* dari *smartphone* ini sangat menonjol karena menyediakan fitur-fitur yang menarik. Hal inilah yang menjadi salah satu faktor kurangnya pembelajaran masyarakat terhadap dasar-dasar teknologi informasi. Masyarakat lebih cenderung memanfaatkan teknologi informasi untuk bermain daripada belajar. Oleh karena itu, dibutuhkan suatu solusi yang dapat membantu mengatasi permasalahan tersebut.

Salah satu cara mengatasi permasalahan di atas yaitu dibuatnya suatu *game* berbasis *android*. Di dalam *game* tersebut dapat dimasukkan unsur pendidikan perangkat teknologi informasi. Namun terkadang sebagian masyarakat dalam bermain suatu *game*, ada masa dimana mereka menemui kesulitan atau *deadlock*. Masyarakat cenderung menyerah atau tidak melanjutkan *game* tersebut jika mengalami *deadlock*. Maka dari itu, dibutuhkan bantuan atau petunjuk untuk permasalahan pada *game* tersebut, yaitu dengan pemberian solusi. Di dalam pemberian solusi tersebut, dibutuhkan suatu algoritma pencarian. Algoritma pencarian adalah algoritma yang menerima sebuah argumen kunci dengan langkah-langkah tertentu dalam mencari rekaman dengan kunci tersebut [1]. Pada dasarnya ada dua teknik pencarian dan pelacakan yang biasa digunakan yaitu pencarian buta (*blind search*) dan pencarian terbimbing (*heuristic search*). Digunakan istilah buta karena memang tidak ada informasi awal yang digunakan dalam proses pencarian [2]. Sedangkan *Heuristik* merupakan sub bidang dari kecerdasan buatan yang digunakan untuk melakukan pencarian dan penentuan rute terpendek [3]. Untuk penelitian ini, digunakan pencarian buta (*blind search*) karena dalam pencariannya tidak dibutuhkan pertimbangan, sedangkan *heuristic* dibutuhkan pertimbangan, jalur mana yang paling baik dalam pencarian.

Pencarian buta (*blind search*) memiliki dua strategi pencarian, yaitu *Breadth First Search* (pencarian melebar) dan *Depth First Search* (pencarian mendalam) [4]. *Breadth First Search* dan *Depth First Search*, keduanya memiliki keuntungan, yang lebih baik bergantung pada masalahnya [5]. Untuk pohon pencarian, *Depth First Search* membutuhkan memori lebih sedikit. Namun, *Depth First Search* dapat terjebak dalam penelusuran yang lama. *Depth First Search* bagus ketika ada banyak kemungkinan solusi, dan kita hanya menginginkan satu saja (tidak peduli yang mana itu). Sedangkan *Breadth First Search* mungkin membutuhkan lebih banyak memori, tapi tidak akan pernah terjebak dan akan selalu menemukan lintasan terpendek yang pertama. Dari perbandingan kedua algoritma itulah, *Breadth First Search* lebih cocok untuk diterapkan dalam *game icon* perangkat teknologi informasi.

## 2. METODE PENELITIAN

*Microsoft Solution Framework* (MSF) adalah pendekatan yang disiplin untuk proyek teknologi berdasarkan serangkaian prinsip-prinsip yang ditetapkan, model, disiplin, konsep, pedoman, dan praktek terbukti. MSF merupakan gabungan dari dua macam model proses, yaitu

---

*waterfall* dan spiral, sehingga dapat dikatakan, MSF merupakan sebuah *Software Development Life Cycle* (SDLC) [6]. Tahapan-tahapan MSF adalah sebagai berikut:

1. *Envisioning*, yaitu tahapan dimana tujuan, manfaat dan ruang lingkup serta batasan dari aplikasi ditetapkan sebagai bentuk awal perencanaan.
2. *Planning*, yaitu tahapan dimana menyiapkan spesifikasi kebutuhan dan perancangan desain aplikasi.
3. *Developing*, yaitu tahapan dimana rancangan yang sebelumnya dibuat, direalisasikan dalam bentuk produk aplikasi.
4. *Stabilizing*, yaitu tahapan dimana dilakukan pengujian pada produk aplikasi yang lengkap dan memperbaiki jika ditemukan kesalahan atau kekurangan.
5. *Deploying*, yaitu tahapan dimana produk aplikasi tersebut digunakan oleh konsumen dan melakukan survey kepuasan konsumen.



Sumber: Microsoft, 2014

**Gambar 1 :Tahapan-tahapan Model Proses MSF**

## 2.1 *Envisioning Phase*

Tahap ini merupakan bentuk awal perencanaan. Pada tahap ini ditentukan tujuan dan manfaat dari pembuatan aplikasi,serta ruang lingkup dari aplikasi *edugame icon* perangkat teknologi informasi yang akan dibuat. *Icon* yang digunakan dalam permainan pada aplikasi ini merupakan gambar *icon software* dan *hardware* komputer.

### 2.1.1 *Software*

*Software* (perangkat lunak) merupakan salah satu komponen penting dalam komputer yang berisi sekumpulan program atau instruksi dalam menjalankan pekerjaannya. *Software* berfungsi sebagai sarana interaksi antara pengguna dengan perangkat keras (*hardware*). Dari segi fisik, *software* tidak dapat dilihat dan disentuh secara langsung, tetapi dapat dioperasikan dengan menggunakan komputer atau media elektronik lainnya.

### 2.1.2 *Hardware*

*Hardware* (perangkat keras) merupakan salah satu komponen dalam komputer yang berfungsi untuk mendukung proses komputerisasi. Berbeda dengan *software*, *hardware* berbentuk nyata, yang dapat dilihat dan disentuh secara langsung. Sistem kerja *hardware* berdasarkan perintah yang telah ditentukan (*instruction set*). Dengan adanya perintah yang dapat dimengerti oleh *hardware* tersebut, maka *hardware* tersebut dapat melakukan berbagai kegiatan yang telah ditentukan oleh pemberi perintah.

## **2.2 Planning Phase**

Pada tahap ini dilakukan analisis sistem dan perancangan desain aplikasi yang akan dibuat. Selain itu, segala macam spesifikasi kebutuhan yang akan digunakan dalam aplikasi juga disiapkan.

### **2.2.1 Use Case Diagram**

*Use Case Diagram* biasanya digunakan untuk menyusun *requirement* dari sebuah sistem. *Use Case Diagram* merupakan pola atau gambaran dari beberapa aktor dan *use case* yang berinteraksi dengan sistem yang akan dibuat. Seorang aktor merupakan entitas manusia untuk melakukan pekerjaan tertentu, sedangkan *use case* merupakan suatu pekerjaan yang dilakukan seorang aktor tersebut.

### **2.2.2 Flowchart**

*Flowchart* adalah rangkaian simbol gambar (*chart*) yang menunjukkan aliran (*flow*) dari proses terhadap data [7]. Simbol-simbol *flowchart* diklasifikasikan menjadi simbol untuk program dan simbol untuk sistem (peralatan *hardware*). *Flowchart* merupakan langkah awal pembuatan program. *Flowchart* disebut juga sebagai cara penyajian suatu algoritma. Dengan menggunakan *flowchart* akan memudahkan dalam menyusun rencana program dan melakukan pengecekan bagian yang terlupakan dalam analisis masalah.

## **2.3 Developing Phase**

Pada tahap ini, mulai direalisasikannya rancangan yang sebelumnya dibuat ke dalam bentuk produk aplikasi dengan menggunakan beberapa perangkat yang sudah disiapkan pada *Planning Phase*.

### **2.3.1 Java 2 Software Development Kit(Java 2 SDK)**

Java 2 SDK meliputi *tool-tool* yang digunakan untuk mengembangkan dan menguji program yang ditulis dalam bahasa pemrograman *Java* dan dijalankan pada *platform Java*. *Tool* didesain untuk digunakan dalam bentuk *command line* seperti interpreter *Java* (*java.exe*), kompiler *Java* (*javac.exe*), *debugger* (*jdb.exe*), dan lainnya [8].

### **2.3.2 Java Development Kit (JDK)**

JDK adalah sebuah produk yang digunakan untuk mengompilasi dan menjalankan program *Java*. JDK berisi sekumpulan dokumentasi dan kode *applet* contoh untuk pengembangan program *Java*. Sejak *Java* diperkenalkan, JDK merupakan *Java SDK (Software Development Kit)* yang paling sering digunakan. Pada tanggal 17 November 2006, *Sun* mengatakan akan merilis JDK sebagai *opensource* menggunakan lisensi *GNU General Public License (GPL)*, artinya *software* ini sekarang menjadi *free software* [9].

### **2.3.3 Android**

*Android* adalah sebuah sistem operasi untuk perangkat mobile berbasis *linux* yang mencakup sistem operasi, *middleware* dan aplikasi [10] *Android* merupakan generasi baru *platform* yang memberikan banyak fasilitas pengembangan yang bebas dan terbuka. Seiring perkembangannya, *Android* telah mengalami pembaruan versi untuk meningkatkan kualitas kinerjanya. Pada aplikasi *edugame icon* perangkat teknologi informasi ini digunakan

---

*android* versi 4.1 (*Jelly Bean*), dengan keunggulan fitur *keybord virtual* yang lebih baik dan pencarian melalui *voice search* lebih cepat.

### 2.3.4 *AndroidSoftware Development Kit (SDK)*

*Android* SDK merupakan *tools* yang digunakan untuk memudahkan programmer dalam mengembangkan aplikasi berbasis *Android*, karena di dalam *Android* SDK sudah ada fungsi-fungsi dasar yang diperlukan untuk membuat program. *Android* SDK berisi *tools Application Programming Interfaces (API)*, sampel, dan dokumentasi dari *Android* yang akan dikembangkan [11]. Untuk memulai membuat emulator *Android*, langkahnya sebagai berikut:

1. Buka SDK Manager.exe pada folder *Android* SDK.
2. Pilih menu *Virtual Device*.
3. Pilih menu *New*.
4. Isi nama emulator pada kotak *Name*.
5. Isi *Android* yang akan digunakan pada kotak target.
6. Isi besar SD Card yang akan digunakan.
7. Isi layar yang akan digunakan pada kolom *Build-in*.
8. Setting *hardware*, kemudian klik tombol *Create AVD* untuk membuat emulator baru.

### 2.3.5 *EclipseInterface Development Environment (IDE)*

*Eclipse* IDE adalah *development tool* yang *portabel* sehingga pengguna tidak harus menginstal aplikasi ini [11]. *Eclipse* merupakan *software free* dan *open source*. *Eclipse* juga menyediakan *plugins* yang dibutuhkan untuk memodifikasi dan membuat aplikasi-aplikasi yang inovatif. Hal itulah yang menjadi alasan mengapa *Eclipse* sangat diminati oleh para pengembang perangkat lunak.

### 2.3.3 *SQLite*

*SQLite* merupakan tempat penyimpanan data yang digunakan secara langsung pada *Android*. *SQLite* merupakan sebuah sistem manajemen basisdata relasional yang bersifat *ACID-compliant* dan memiliki ukuran *library* yang relatif kecil. *Software* ini merupakan produk *public domain* dan ditulis menggunakan bahasa C [9].

## 2.4 *Stabilizing Phase*

Pada tahap ini dilakukan pengujian pada aplikasi lengkap yang telah dibuat sebelum produk dikenalkan kepada pengguna atau konsumen. Pengujian dilakukan untuk menemukan kesalahan dan memperbaiki kekurangan yang terdapat pada aplikasi.

## 2.5 *Deploying Phase*

Pada tahap ini, pengujian produk aplikasi sudah selesai. Aplikasi mulai dapat digunakan oleh pengguna atau konsumen, sekaligus mendapat kritik dan saran dari pengguna terhadap aplikasi yang telah dibuat.

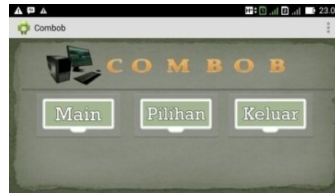
### 3. HASIL DAN PEMBAHASAN

#### 3.1 Tampilan Antarmuka Aplikasi

Pada prosedur uji coba aplikasi akan dijelaskan tentang hasil jadi *edugame icon* perangkat teknologi informasi berbasis *android* serta fungsi setiap tampilan agar pengguna dapat menggunakan aplikasi ini secara maksimal.

##### 3.1.1 Tampilan Menu Utama

Tampilan Menu Utama (Gambar 2) merupakan tampilan awal ketika aplikasi dijalankan. Di dalam halaman ini terdapat 3 *button* menu, yaitu Main, Pilihan dan Keluar. Untuk memulai permainan, dapat ditekan *button* Main, sedangkan untuk keluar dari permainan, dapat ditekan *button* Keluar. *Button* Pilihan berfungsi untuk melihat skor tertinggi permainan, melihat petunjuk permainan, serta mengatur suara.



Gambar 2 : Tampilan Menu Utama

##### 3.1.2 Tampilan Level 1

Tampilan Level 1 (Gambar 3) merupakan tampilan permainan level 1 ketika *button* Main pada Menu Utama (Gambar 2) dipilih. Cara mainannya adalah cari 2 *icon* yang sama atau berpasangan. Jika kedua *icon* yang dipilih cocok, maka akan muncul tampilan soal tentang *icon* yang sudah dipilih tersebut. Pada level 1 ini, pemain harus dapat memecahkan 10 soal dalam waktu 300 detik untuk melanjutkan ke level 2. Jika pemain belum memecahkan 10 soal dan waktu habis, maka permainan berakhir.

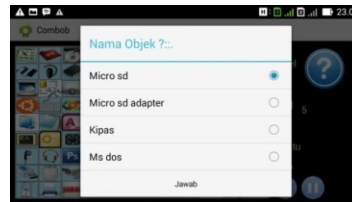


Gambar 3 : Tampilan Level 1

##### 3.1.3 Tampilan Soal

Tampilan Soal (Gambar 4) muncul ketika kedua *icon* yang dipilih cocok. Pemain harus memilih satu dari empat pilihan jawaban. Jika jawaban benar, permainan pencarian *icon* dilanjutkan. Jika jawaban salah, permainan juga dilanjutkan, tetapi posisi *icon* akan teracak.

---



**Gambar 4 : Tampilan Soal**

### 3.1.4 Tampilan Bantuan

Tampilan Bantuan (Gambar 5) muncul apabila pemain mengalami *deadlock* dalam mencari *icon* yang cocok. Tekan tombol tanda tanya, kemudian sistem akan menunjukkan kedua *icon* yang cocok dengan memblok kedua *icon* tersebut. Pada bantuan inilah yang menunjukkan penerapan algoritma *Breadth First Search* (BFS) pada aplikasi.



**Gambar 5 : Tampilan Bantuan**

### 3.1.5 Tampilan Level 2

Tampilan Level 2 (Gambar 6) merupakan tampilan permainan level 2 ketika level 1 (Gambar 3) terselesaikan. Cara permainannya sama dengan level 1, hanya perbedaannya terletak pada ketentuan permainan. Pada level 2 ini, pemain harus dapat memecahkan 15 soal dalam waktu 300 detik untuk melanjutkan ke level 3 (Gambar 7). Jika pemain belum memecahkan 15 soal dan waktu habis, maka permainan berakhir.



**Gambar 6 : Tampilan Level 2**

### 3.1.6 Tampilan Level 3

Tampilan Level 3 (Gambar 7) merupakan tampilan permainan level 3 ketika level 2 (Gambar 6) terselesaikan. Cara permainannya sama dengan level-level sebelumnya, hanya perbedaannya terletak pada ketentuan permainan. Pada level 3 ini, pemain harus dapat memecahkan 20 soal dalam waktu 300 detik untuk melanjutkan ke level 4 (Gambar 8). Jika pemain belum memecahkan 20 soal dan waktu habis, maka permainan berakhir.

---



**Gambar 7 : Tampilan Level 3**

### **3.1.7 Tampilan Level 4**

Tampilan Level 4 (Gambar 8) merupakan tampilan permainan level 4 ketika level 3 (Gambar 7) terselesaikan. Cara mainannya sama dengan level-level sebelumnya, hanya perbedaannya terletak pada ketentuan permainan. Pada level 4 ini, pemain harus dapat memecahkan 25 soal dalam waktu 300 detik untuk melanjutkan ke level 5 (Gambar 9). Jika pemain belum memecahkan 25 soal dan waktu habis, maka permainan berakhir.



**Gambar 8 : Tampilan Level 4**

### **3.1.8 Tampilan Level 5**

Tampilan Level 5 (Gambar 9) merupakan tampilan permainan level 5 ketika level 4 (Gambar 8) terselesaikan. Level 5 merupakan level terakhir dari permainan. Cara mainannya sama dengan level-level sebelumnya, hanya perbedaannya terletak pada ketentuan permainan. Pada level 5 ini, pemain harus dapat memecahkan 30 soal dalam waktu 300 detik untuk mengakhiri permainan. Jika pemain belum memecahkan 30 soal dan waktu habis, maka permainan berakhir.



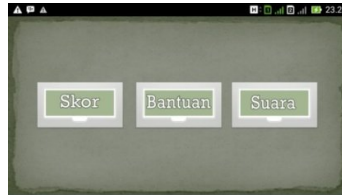
**Gambar 9 : Tampilan Level 5**

### **3.1.9 Tampilan Menu Pilihan**

Tampilan Menu Pilihan (Gambar 10) merupakan tampilan ketika *button* Pilihan dipilih pada Menu Utama (Gambar 2). Halaman ini terdapat 3 *button* menu, yaitu Skor, Bantuan dan Suara. Menu Skor (Gambar 11) untuk melihat catatan skor tertinggi pemain. Menu Bantuan (Gambar 12) untuk melihat cara bermain dari aplikasi. Menu Suara (Gambar 13) untuk mengaktifkan atau menonaktifkan suara permainan.

---





**Gambar 10 : Tampilan Menu Pilihan**

### 3.1.10 Tampilan Menu Skor

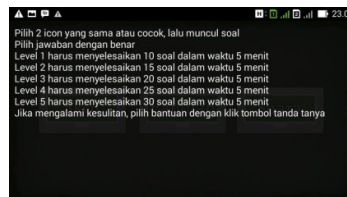
Tampilan Menu Skor (Gambar 11) merupakan tampilan ketika *button* Skor dipilih pada Menu Pilihan (Gambar 10). Pemain dapat melihat catatan skor tertinggi yang pernah dicapai.



**Gambar 11 : Tampilan Menu Skor**

### 3.1.11 Tampilan Menu Bantuan

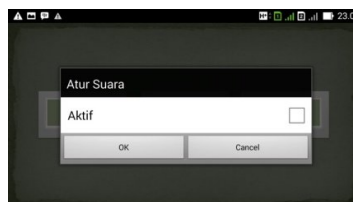
Tampilan Menu Bantuan (Gambar 12) merupakan tampilan ketika *button* Bantuan dipilih pada Menu Pilihan (Gambar 10). Pemain dapat melihat petunjuk cara bermain dari aplikasi.



**Gambar 12 : Tampilan Menu Bantuan**

### 3.1.12 Tampilan Menu Suara

Tampilan Menu Suara (Gambar 13) merupakan tampilan ketika *button* Suara dipilih pada Menu Pilihan (Gambar 10). Pemain dapat mengatur, apakah ingin mengaktifkan atau menonaktifkan suara permainan.



**Gambar 13 : Tampilan Menu Suara**

---

### 3.2 Analisis Hasil Pengujian Permainan

Pengujian dilakukan untuk mengetahui apakah aplikasi dapat berjalan dengan baik sesuai dengan tujuan pada awal perencanaan. Berikut analisis hasil pengujian pada aplikasi *edugame icon* perangkat teknologi informasi berbasis *android*:

#### 3.2.1 Analisis Pengujian Instalasi

Pengujian Instalasi merupakan tahap pengujian awal dari aplikasi. Aplikasi diinstal dan dapat dijalankan pada *smartphone*.

**Tabel 1 Analisis Pengujian Instalasi**

Operasi	Kondisi	Hasil
Instalasi aplikasi	Aplikasi dapat dijalankan pada <i>smartphone</i> .	Berhasil.

#### 3.2.2 Analisis Pengujian Menu Utama

Pengujian Menu Utama merupakan tahap pengujian dari tampilan Menu Utama aplikasi. *Button* menu (Main, Pilihan dan Keluar) yang ada pada tampilan Menu Utama (Gambar 2), diuji satu per satu, apakah sudah berjalan sesuai fungsinya. Hasil dari pengujian *button* menu ketiganya sudah berhasil.

**Tabel 2 Analisis Pengujian Menu Utama**

Operasi	Kondisi	Hasil
Klik tombol main	Sistem menampilkan tampilan level 1 permainan dan dapat langsung dimainkan.	Berhasil.
Klik tombol pilihan	Sistem menampilkan tampilan menu pilihan, dimana pemain dapat melihat dan mengatur permainan.	Berhasil.
Klik tombol keluar	Sistem keluar dari aplikasi yang berjalan.	Berhasil.

#### 3.2.3 Analisis Pengujian Level Permainan

Pengujian Level Permainan merupakan tahap pengujian dari tampilan keseluruhan level permainan. Pada tahap pengujian ini ditemukan beberapa kesalahan-kesalahan yang terjadi, diantaranya sistem tidak dapat mendeteksi kecocokan *icon* yang telah dipilih. Kesalahan berikutnya, ketika *icon* pasangan terpilih, soal tidak muncul. Kesalahan juga terjadi ketika menjawab soal. Jawaban soal sudah benar, tetapi sistem memberitahukan bahwa jawaban salah, dan posisi *icon* tidak teracak ketika jawaban tersebut salah. Untuk tombol-tombol seperti bantuan dan *pause*, sudah berjalan sesuai fungsinya.

**Tabel 3 Analisis Pengujian Level Permainan**

Operasi	Kondisi	Hasil
Pencocokan <i>icon</i>	Pilih <i>icon</i> pertama yang diinginkan, kemudian pilih <i>icon</i> kedua yang sama atau merupakan pasangan dari <i>icon</i> pertama. <i>Icon</i> pertama yang	Terjadi <i>error</i> . Sistem tidak dapat mendeteksi kecocokan <i>icon</i> kedua

	sudah dipilih, tidak dapat dipilih ulang. Artinya <i>icon</i> pertama akan menunggu <i>icon</i> kedua dipilih. Jika <i>icon</i> kedua yang dipilih bukan pasangan <i>icon</i> pertama, <i>icon</i> pertama kembali seperti semula.	yang dipilih dengan <i>icon</i> pertama.
Muncul soal	Pilih <i>icon</i> pertama yang diinginkan, kemudian pilih <i>icon</i> kedua. Kedua <i>icon</i> yang dipilih tersebut harus sama atau cocok.	Terjadi error. Soal tidak muncul, padahal kedua <i>icon</i> sudah cocok.
Jawab soal	Pilih jawaban dari soal yang muncul. Jika jawaban benar, sistem akan menampilkan kembali ke permainan <i>icon</i> . Jika salah, sistem juga menampilkan kembali ke permainan <i>icon</i> , akan tetapi posisi semua <i>icon</i> akan teracak. Kondisi kotak <i>icon</i> yang sudah terjawab benar akan kosong.	Terjadi error. Jawaban yang dipilih sudah benar, namun sistem memberitahukan bahwa jawaban salah. Posisi <i>icon</i> juga tidak teracak ketika jawaban salah.
Pilih bantuan	Klik tombol bantuan. Sistem akan menunjukkan posisi <i>icon</i> yang berpasangan.	Berhasil.
Pilih <i>pause</i>	Klik tombol <i>pause</i> . Waktu permainan akan berhenti sementara. Klik kembali tombol <i>pause</i> untuk melanjutkan permainan.	Berhasil.

### 3.2.4 Analisis Pengujian Menu Pilihan

Pengujian Menu Pilihan merupakan tahap pengujian dari tampilan Menu Pilihan. *Button* menu (Skor, Bantuan dan Suara) yang ada pada tampilan Menu Pilihan (Gambar 10), diuji satu per satu, apakah sudah berjalan sesuai fungsinya. Hasil dari pengujian *button* menu ketiganya sudah berhasil.

**Tabel 4 Analisis Pengujian Menu Pilihan**

Operasi	Kondisi	Hasil
Klik tombol skor	Sistem menampilkan tampilan skor tertinggi yang pernah dicapai pemain.	Berhasil.
Klik tombol bantuan	Sistem menampilkan tampilan petunjuk permainan.	Berhasil.
Klik tombol suara	Sistem menampilkan tampilan pilihan untuk mengaktifkan atau menonaktifkan suara.	Berhasil.

## 4. KESIMPULAN

Dalam pengembangan aplikasi *edugame icon* perangkat teknologi informasi berbasis *android* ini telah melalui beberapa tahap, mulai dari tahap awal perencanaan, perancangan desain, implementasi, serta tahap pengujian, hingga dapat ditarik kesimpulan sebagai berikut:

1. Algoritma *Breadth First Search* (BFS) dapat memberikan solusi dalam proses pencarian *icon*, ketika pemain mengalami kesulitan dalam menemukan *icon* yang berpasangan.
2. Aplikasi ini dibuat untuk mengenalkan masyarakat terhadap nama perangkat lunak (*software*) dan perangkat keras (*hardware*) beserta fungsinya.

## 5. SARAN

Beberapa saran yang dapat digunakan dalam pengembangan aplikasi ini, antara lain sebagai berikut:

1. Pada pengembangannya, aplikasi ini diharapkan mendapat penambahan fitur, seperti nama pemain yang dicantumkan dalam skor, efek suara ketika *icon* disentuh, dan lain sebagainya sesuai dengan kebutuhan pengguna.
2. Pada pengembangannya, diharapkan *icon* perangkat teknologi informasi lebih update dan mendapatkan penambahan jumlah.
3. Pada level 2 hingga level 5, dengan banyaknya jumlah soal yang harus diselesaikan, waktu permainan 5 menit terasa singkat. Diharapkan pada pengembangannya, waktu permainan dapat ditambah sekitar 7 menit hingga 10 menit.

## DAFTAR PUSTAKA

- [1] Novriyanto, ST., M.Sc, & S, Zaid, M., ST, 2013, *Penerapan Algoritma Backtracking Berbasis Blind Search untuk Menentukan Penjadwalan Mengajar*, Universitas Islam Negeri Sultan Syarif Kasim Riau, Pekanbaru
  - [2] Suyanto, 2007, *Artificial Intelligence*, Informatika, Bandung
  - [3] Alamsyah, 2010, *Pemanfaatan Metode Heuristik Dalam Pencarian Minimum Spanning Tree dengan Algoritma Semut*, Universitas Tadulako, Palu
  - [4] Desiani, Anita, & Arhami, Muhammad, 2006, *Konsep Kecerdasan Buatan*, Andi, Yogyakarta
  - [5] Inggiantowi, Hafid, 2008, *Perbandingan Algoritma Penelusuran Depth First Search dan Breadth First Search pada Graf serta Aplikasinya*, Institut Teknologi Bandung, Bandung
  - [6] Gautama, Surya, Ryandi, et al, *Rancang Bangun Aplikasi Pembelajaran Bahasa Inggris Berbasis Multimedia Pada Bimbel Excellence*, STMIK GI MDP, Palembang
  - [7] Suarga, 2006, *Algoritma Pemrograman*, Andi, Yogyakarta
  - [8] Foenadioen, 2008, *Pengembangan Aplikasi Web Database Menggunakan Java Server Page*, Andi, Yogyakarta
  - [9] P, Hernita, 2013, *Programming with Eclipse*, Andi, Yogyakarta
  - [10] Murtiwiayati, & Lauren, Glenn, 2013, *Rancang Bangun Aplikasi Pembelajaran Budaya Indonesia untuk Anak Sekolah Dasar Berbasis Android*, Universitas Gunadarma, Depok
  - [11] Arifianto, Teguh, 2011, *Membuat Interface Aplikasi Android Lebih Keren dengan LWUIT*, Andi Offset, Yogyakarta
-